NOSC TD 1199

NAVAL OCEAN SYSTEMS CENTER San Diego, California 92152-5000

Technical Document 1199
November 1987

Distributed Tactical Decision Support By Using Real-Time Database System

Dana L. Small

Approved for public release; distribution is unlimited.

NAVAL OCEAN SYSTEMS CENTER

San Diego, California 92152-5000

E. G. SCHWEIZER, CAPT, USN
Commander

R. M. HILLYER
Technical Director

ADMINISTRATIVE INFORMATION

The work reported here was performed by members of the C^2 Information Processing and Display Technology Branch, Advanced C^2 Technologies Division, with funding provided by the Office of Naval Research. Development of new track support logic was supported by Lui Sha and H. Tokuda, of Carnegie-Mellon University, under Navy Contract N0001487-WX-24122. The China Sea Scenario was developed by M. Middleton, of System Development Corporation, under Navy contract N66001-83-D-0094. This research is sponsored by the Office of Naval Research's Program in Distributed Tactical Decision Making, Contract Number N0002487-WX-24122.

Released by
E.J. Wells, Head (Acting)
C^2 Information Processing
and Display Technology Branch

Under authority of
W.T. Rasmussen, Head
Advanced C^2
Technology Division

ACKNOWLEDGMENTS

The author wishes to thank Lui Sha of Carnegie Mellon University for his insight into the application of his Modular Concurrency Control (MCC) Theory; David Schmidt of the NOSC System Design Branch for his intensive help in the analysis of the combat system database used for this work; Mike Pohoski of the NOSC Software Quality Assurance Branch for his insight into combat direction system tracking; and Marsh Middleton, System Development Corporation, for his insight into the development of the China Sea Scenario.

RH

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release, distribution is unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
NOSC TD 1199

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Naval Ocean Systems Center

6b. OFFICE SYMBOL (if applicable)
Code 443

7a NAME OF MONITORING ORGANIZATION

6c. ADDRESS (City, State and ZIP Code)
San Diego, CA 92152-5000

7b. ADDRESS (City, State and ZIP Code)

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
Office of Naval Research

8b. OFFICE SYMBOL (if applicable)
Code 1223

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
N0001487-WX-24122

8c. ADDRESS (City, State and ZIP Code)
Arlington, VA 22217

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	AGENCY ACCESSION NO.
62721N	RS21241	ONR	DN488 752

11. TITLE (include Security Classification)
Distributed Tactical Decision Support By Using Real-Time Database System

12. PERSONAL AUTHOR(S)
Dana L. Small

13a. TYPE OF REPORT
Interim

13b. TIME COVERED
FROM TO

14. DATE OF REPORT (Year, Month, Day)
November 1987

15. PAGE COUNT
38

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Computers, subsystems, command control (C^2), distributed processor architecture, distributed tactical decision making (DTDM), modular concurrency control (MCC) theory, distributed tactical decision support tracking models

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This interim report describes findings ~~on the~~ of applications of ONR's distributed tactical decision making (DTDM) research performed by Carnegie-Mellon University ~~(CMU)~~ in modular concurrency control (MCC) theory, using a distributed tactical decision support tracking model created by NOSC. Initial results in the application of MCC theory and a short analysis of the reasoning used is developed. In addition, an architecture now being developed for a distributed real-time database version of the processing for the tracking model is discussed. Finally, plans for other possible experiments are detailed.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
□ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT □ DTIC USERS

21 ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
Dana L. Small

22b. TELEPHONE (include Area Code)
(619) 553-4130

22c. OFFICE SYMBOL
Code 443

83 APR EDITION MAY BE USED UNTIL EXHAUSTED
ALL OTHER EDITIONS ARE OBSOLETE

CONTENTS

FIGURES

INTRODUCTION

The goal of this document is to report on:

1. The NOSC evaluation of basic research results relating to ONR's Distributed Tactical Decision Making (DTDM) program in real-time distributed database management, modular concurrency control, and maintenance of consistent views of information through experimentation in our tactical C^2 prototype testbed.
2. Our investigation into real-time distributed data management with respect to recovery from processor failure or from an intentional interruption of processing.
3. The assurance of timeliness, availability of data, and correct analysis of the arrival of time-latent data in such a real-time database system.

What follows is an interim report on the application of DTDM research performed by Carnegie-Mellon University (CMU) in modular concurrency control (MCC) theory (reference 1), using a distributed tactical decision support tracking model created by NOSC. Initial results in the application of MCC theory and a short analysis of the reasoning used will be developed. In addition, an architecture we are developing for a distributed real-time database version of the processing system for such a tracking model will be discussed. Finally, plans for other possible experiments will be detailed.

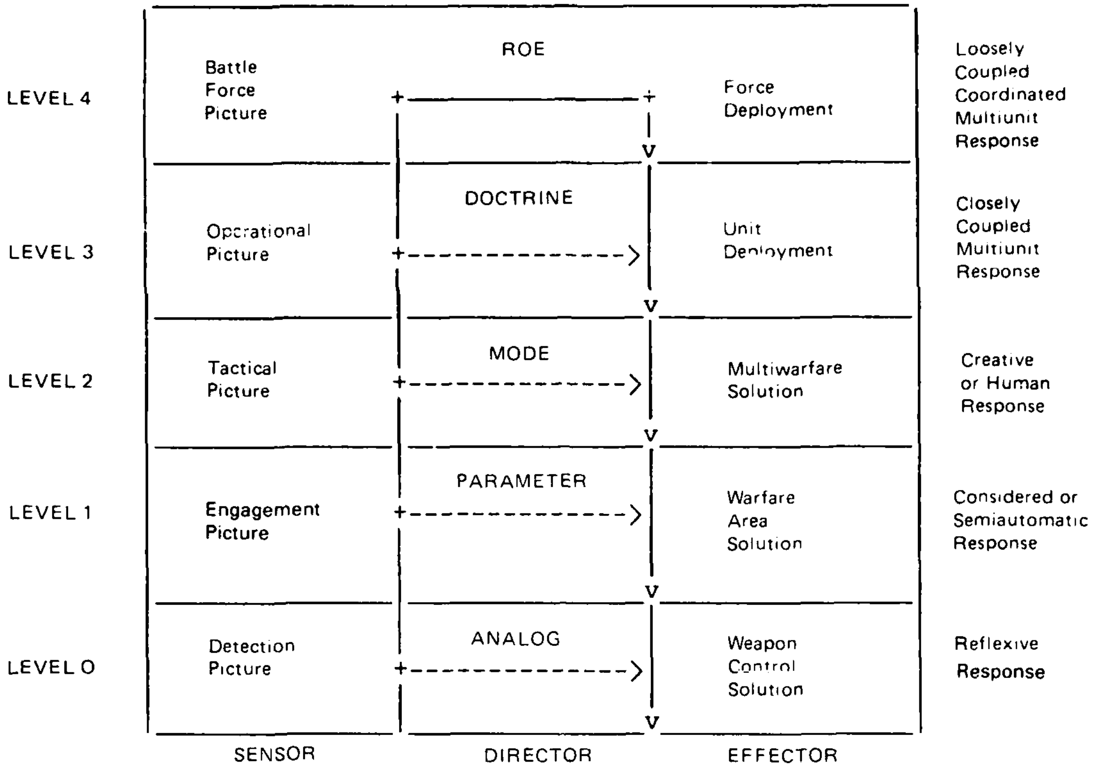
DISTRIBUTED TACTICAL DECISION SUPPORT MODEL

Figure 1, from reference 2, gives an overview of a possible Navy C^2 information system, indicating the types of data elements used. The data object model was used to aid in formally specifying rules and the results of rules governing which of those data elements should be communicated and when communication should occur between levels and within levels. Because data elements can be aggregated at the higher levels of figure 1, it is likely that responsiveness can be maintained. The desired result of using the model would be to specify minimal communications and maximum concurrency, taking into account these two observations (see reference 2).

Using this model

1. Passive components, called objects, are developed as tracks.
2. Operations for initiating state changes of objects, such as noting a change in the identification of a track or track movement, are denoted.
3. A dependency representation scheme between operations on different objects, such as maintenance of a consistent view of a track whose data values can come from a variety of different sources*, is developed.

*The scheme will ensure that a copy of a track is maintained at the engagement level when its identity was noted to be hostile at the battle force level. The focus will be on the possibility of representation by using the mechanisms of CMU's modular concurrency control (MCC) scheme (reference 1), together with a means of maintaining consistent views of information by using the catalog directory proposed in this report and in the author's paper, *Distributed Real-Time Database System for Command/Control and Combat Systems* (reference 3).



ROE=Rules of Engagement

Figure 1. Layered conceptual model for Navy C2 (reference 1).

4. A formal interface description for data or functional abstractions between echelons (or layers) and protocols among operations within a layer are suggested, with no elaboration.

This report focuses on the development of concepts 1 through 3 at the reflexive response layer of figure 1. These concepts probably could be applied to the interfaces, but this thought will not be developed further in this report.

At the reflexive response or weapon control layer, the decision support model is based on the birth/death of a contact, which is designated as a track number (data object), together with time of entry. The data values for the contact are determined whenever it is first established by own-ship sensor or by a remote source. The model will be capturing different views of the contact as it enters via the various processes of the CV's combat direction system (CDS). The transactions used in the model to capture these views are described at a high level in Richard Kauffold's Naval Postgraduate School thesis, *The Entity-Relationship Approach: A Good Tool for Tactical Data Systems?* (reference 4). The system management of these views in the tracking model will be based in part on the combat direction system as synopsized in appendix A and detailed in depth in the Advanced Combat Direction System Specification (reference 5).

The assumption is that Time 0 (T0) of any contact (or its "birth") for this particular model is when the contact first enters the similar source integration (SSI) building block of the CV's combat direction system (CDS), or when a remote data process (RDP) or an afloat correlation system (ACS) contact source enters the dissimilar source integration (DSI) function and its track number is established. In the diagram on page 50 of reference 4, T0 occurs when the sensor gains contact and the track is formed. The "death" of the contact could occur because of the loss of sensor contact or because of contact engagement, and in both cases can carry severe real-time constraints. In the case of engagement and the monitoring that ensues, the ouput of the CV CDS, correlated identified track(s), would be sent via the multiwarfare control building block for the processing of weapon engagement, direction, or aircraft orders.

The basic system building blocks are as shown in figure 2. They include the similar source integration (SSI) functions for radar/IFF (identification friend/foe); acoustic and navigation ESM (electronic support measures); the remote data process (RDP) function; the afloat correlation system (ACS); the dissimilar source integration (DSI) function; the database function, including the battle force track file (BFTF); the multisource identification (MSID) function; the user functions for correlated track data; multiwarfare control (MWC); and mission control (MC). It is assumed that all basic system building blocks can participate in the establishment, engagement, and/or loss of the contact; i.e., the contact decision process. However, there will be no attempt in this model to provide manual adjustments for sensor performance, such as those included in the surveillance control function. Navigation data will be assumed to be correct from the combination of Joint Tactical Information Distribution System (JTIDS) and own-ship navigation data. All building blocks will have computer consoles with human operators that participate in the contact decision process. The model will concentrate on real-time constraints encountered in the tracking process during determination of a track's threat potential.

Data will be gathered on performance by using the China Sea Scenario (reference 7), which is presented in appendix B. It is an unclassified scenario used to evaluate mathematical correlation techniques for multisource track control at the combat direction system level. With the scenario, we established the contacts that were made and ran an analysis on a centralized computer system with a single operator. In a sense, it is a ground truth, from which we can vary the number of operators, distribute the decision making, and analyze the results of alternative methods for distribution. Appendix B of reference 7 and appendix A of this report are detailed descriptions that discuss the role of the various sensors (i.e., when detections are made), when aircraft are launched, and the role of the participating platforms (i.e., position, identification, velocity). Once the tracking model is implemented by using this scenario, and the system building blocks (as listed above and described in more detail in appendix A and reference 5) come into use, a number of relative performance measurements can be taken. They will be a function of concurrency control decisions; e.g., when to maintain consistent views of data, and how current they should be. The measurements will take the form of the cost of providing timeliness of data; the cost of reversing the decision; the cost of missing schedule deadlines, where cost could equate to time and quality of data (i.e., the correctness of the data); and the validity of the distributed decision making (i.e., based on the percentage of correct decisions made as a function of time).

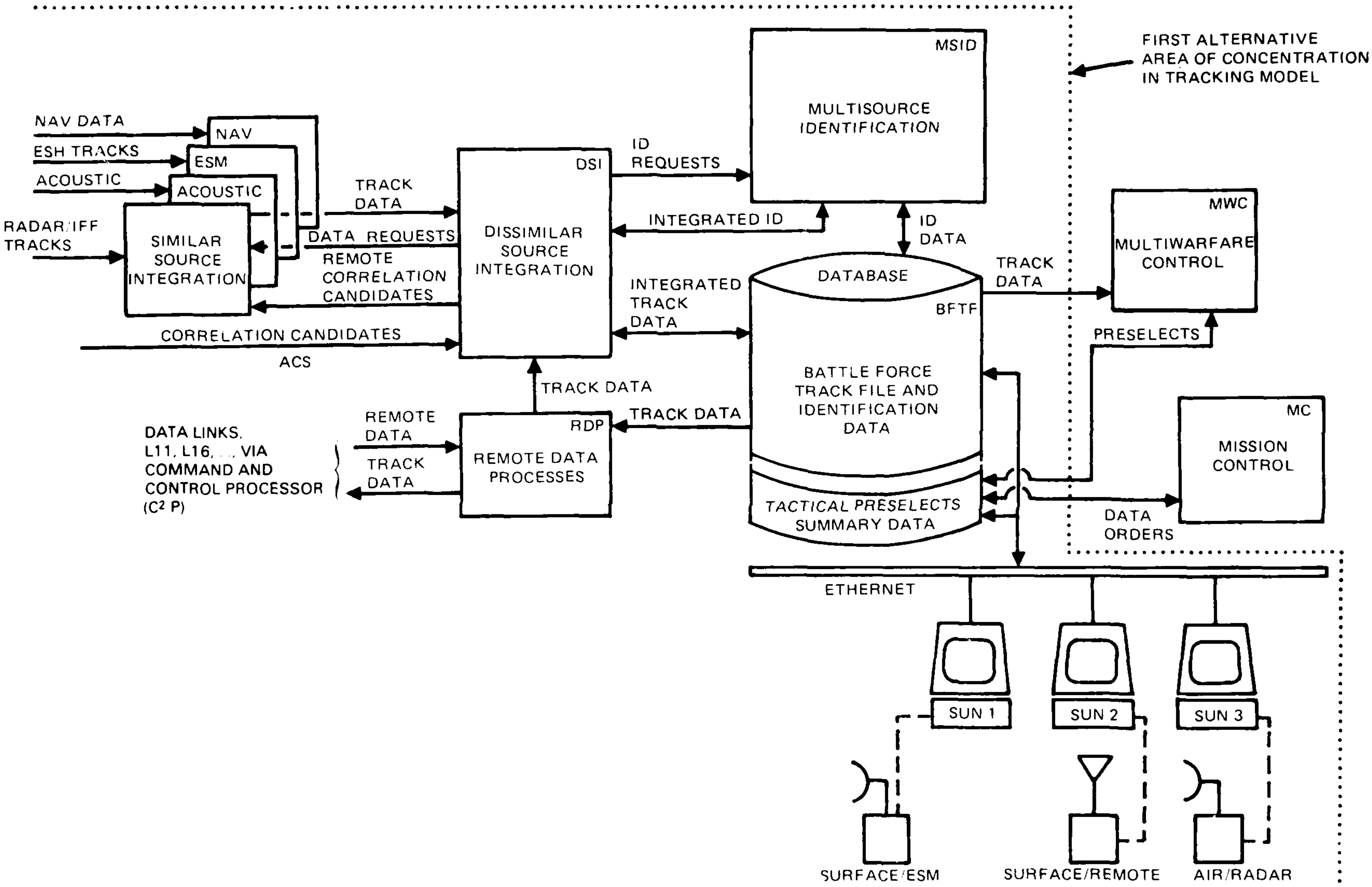


Figure 2. DTDM initial configuration.

TRACKING MODEL EXPERIMENT

All of the actions provided in this model are in support of transactions that involve the establishment of a contact by a sensor, a remote data process (RDP), or an afloat correlation system (ACS) data source; and the contact's engagement, its loss, or its reassessment as a contact with a different value. Time 0 for any given transaction will be at the time a contact enters a similar source integration (SSI) function, or when an ACS or an RDP contact data source enters the dissimilar source integration (DSI) function. In the context of the China Sea Scenario (appendix B), that contact will be assigned a track number with at least a time of entry and the type of sensor or data source that established the contact. Each of these functions will determine whether the track is unique within the similar sources available to it. In all instances, these functions will assess the track's location and its type; its name, if available; and its ID. The radar local track processing data are likely to contain a better concept of the track's location. ESM local track processing is likely to provide good insight as to track emissions and possible location data if data are available from more than one source. The acoustic SSI information has a slower arrival rate; uses the same type of data as does ESM local track processing, but for subsurface contacts; and is of lower quality because it is more ambiguous (the acoustic SSI function and the ACS function are not modeled in this use of the China Sea Scenario). The ACS function will provide correlation recommendations between real-time and nonreal-time tracks. The RDP function provides a coordinated track input from remote data sources; i.e., other platforms. In this version of the model, the output of each of the local track, SSI, ACS, and RDP processes will include the track number; the time of arrival of the track; its latitude, longitude, altitude or depth, course, speed, type, and name if available; identification; emissions; type of sensor or data source; and quality of track data. Track number and time of arrival are the only entries that must be filled by the CDS system. The others may or may not be filled, depending on the sensor's or that data source's abilities. The local track processing, SSIs, ACS, and RDP will include a model of time, data quality (i.e., sensor operability, which could include loss of contact; operator's opinion; or percentage of confidence in the data provided), and where the data are to be sent next. All data from each of these functions will be sent to the battle force track file (BFTF) database function and possibly to predetermined DSI operator consoles.

Track information will be received from applicable SSI functions and input to the centralized computer database resident in the BFTF database function. The receipt of track information by the DSI consoles is filtered by the CDS system, according to what the operator needs. The operator's information may be sent according to what the doctrine says he should be monitoring. Track information also may be required if the commanding officer requests more information in his attempt to characterize a track as threatening or not. The operator also may have the leeway to decide what is most interesting. Or he may have little choice, such as when an unanticipated threat arrives. The analysis that follows shows how information is passed between operators, how long it takes for threat analysis, and the quality of that analysis for various distribution options of operator consoles.

The material in the following four paragraphs is adapted from reference 6. Initially, consoles are assigned processing responsibility by console, such as radar, ESM, and remote sensor and data source types; and for the contact types surface, subsurface, and aircraft by sector areas. Each console can be considered to be working serially and independently on its own set of tracks to make a determination of identity and location. There is no guarantee that the tracks are separate and distinct between consoles. Likewise, there is no guarantee that a track being watched on one console is received at the same time by the CDS system as when it is monitored at another console. An example of such serialized processing could consider each console having a short track history, so that the operator would receive a track report and match it with one of the tracks available in his console's track history. He would assess whether the track had its fire-control radar on, or an indication of any other threatening maneuver such as change of course and speed, a communication break, or radio silence. He would then send information back to the main computer, or other consoles with an interest in the track, as to whether the track is authentic. Track data would be kept internally consistent by maintaining constraint rules. Examples of these rules could be a comparison of track speed with maximum air contact speeds to assess what the track could or could not be (e.g., not a helicopter because its speed is too great); or how much maneuverability is allowed for various platforms (e.g., how much acceleration can be assumed reasonably from the changing velocity of that platform, how quick its course can be changed, etc.). Serialized analysis of a track's location and identification can be interrupted upon receipt of a new 'very important' track, such as a new track believed to be hostile, which MUST be included in the console's local database. In this kind of instance, the ongoing analysis would be halted and rolled back to a recoverable start point.

A console's view can be formed of tracks that might lead to inconsistency from another console's view. For example, the aircraft console's view of an air contact's identification could indicate it to be hostile, whereas the ESM console's view of that same contact could see it as being friendly. These views can become consistent for a number of different reasons, while they still obey the constraint rules established when the consoles are operating independently. First, the console operator may receive new information on the track as it is broadcast from the centralized BFTF database. Second, the centralized computer could receive an operator alert to a track he feels to be hostile, which is then sent by the computer to all consoles asking their opinions. Third, it could be that a console or group of consoles becomes overloaded with responsibility for too many tracks, so that the responsibility would be shifted to less lightly loaded consoles, forcing data consistency to be resolved when they change console locations. Or the main computer could change a console's responsibilities because of a change in perceived threat, such as noting a close target and assigning responsibility to an appropriate console in a nearby sector. Or console operators, either by doctrine or by direction from the tactical action office (TAO) and or the commanding officer, are told they must report to him all hostile identifications within a preselected radius of own-ship, forcing a consensus view of differing opinions.

A second alternative (see figure 3) would be that of minimizing the global track processing by only performing correlation processing on new tracks or on those tracks which have radical changes. A new report arriving will be a candidate for a 'global track' if its quality exceeds a certain confidence level threshold. Once it exceeds that threshold level, it has either established an association with other local-level tracks via correlation or, vacuously, has become a global track if no association has yet been established. The model consists of two levels for this alternative, the local level (combining the local sensor level with the similar source integration level) and the dissimilar source integration (DSI) level. The local level has its own local track database and a global track database (global track numbers for local tracks). If a new report already has a global track number and there are no significant changes, local and global history will be updated and correlation will not be attempted at the DSI level. The DSI level consists of a global track database and 'candidates' for a global track. If a 'candidate' passes the correlation criteria, it is given a unique global track number, which is propagated downwards to the local level (inserted in their global track databases). The theory is based on the assumption that a track is an atomic data set (ADS, see reference 1), the smallest data unit that can be synchronized. Elementary transactions on such ADSs include local constraint and association transactions and global track correlation (see appendix F for descriptions of the new track report logic that is provided in the consoles for both alternatives). In all cases, the execution of these transactions must preserve the constraint or consistency rules established on

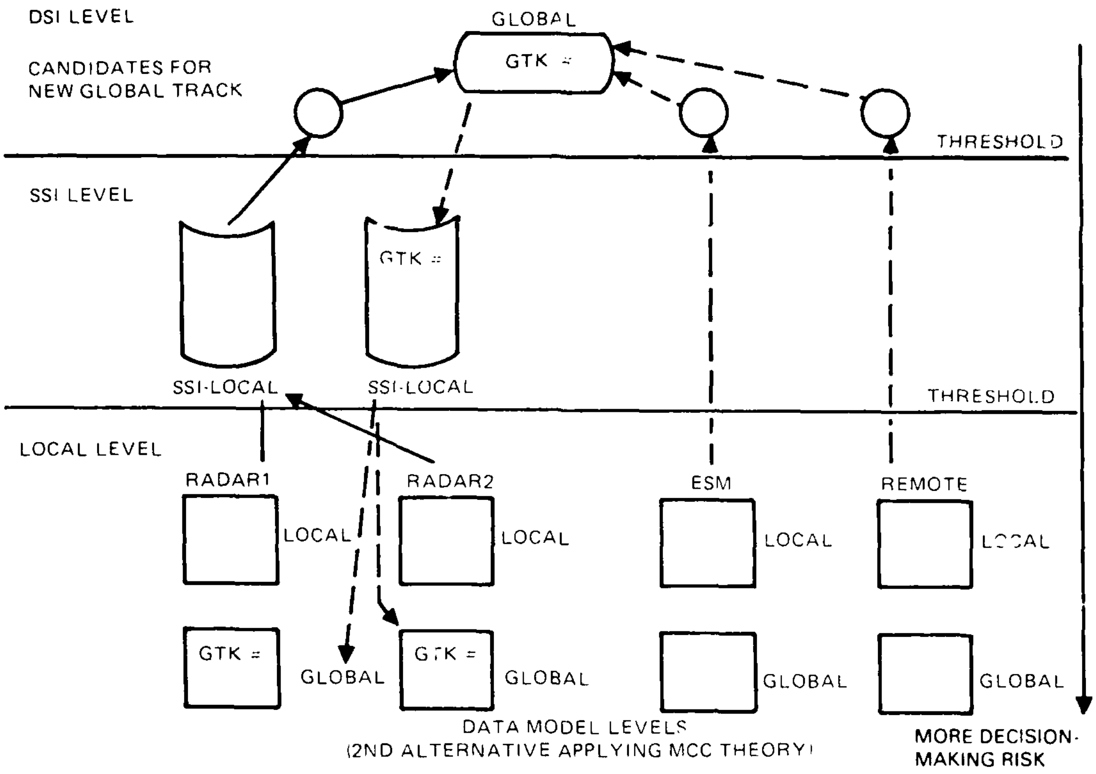


Figure 3. Distributed tactical decision support.

entry of the track into the SSI DSI processing levels to ensure correctness. For example, a track's correlation processing cannot be interrupted without preserving its state before the correlation started, therefore preserving the correctness of the transaction.

In both alternatives, when a track is agreed to be hostile, the next step will be the sending of its value to the multiwarfare control (MWC) and mission control (MC) building blocks. The MWC building block will support the action of weapon assignment and subsequent scheduling and engagement. The MC building block supports among other things the information needs of users such as the flag data display system, which resides at the warfare area level, for a complete accurate assessment of the tactical situation. Operator consoles, in both cases, will be modeled only as to time to assess the threat and the validity of that assessment. Engagement will be considered the "death of a contact" and an end time will be assigned to that track.

The China Sea Scenario (see Appendix B) is the environment within which these experiments are continuing to be performed. Data are being taken based on a determined objective, such as how much time is used in assessing the magnitude of the threat or reacting to a change in threat status; or when a sector of operations should be handed over to another console; or how large a scale of operations can be handled by a console or group of consoles. Implicit in all of these will be an analysis of the concurrency of actions taken as well as the accuracy or validity of the decision made (i.e., was the threat assessment correct?).

Our initial findings are based on a first assessment of how well concurrency is working in such a scenario. The data were taken over a series of 50 reports for 648 simulated seconds, from a potential of 370 reports over a period of 1368 simulated seconds (see figure 4). The configuration for the tracking model is shown in figure 5. In this first experiment, only the local radar process, the remote process, and the estimated-time-of-contact global-kinematic

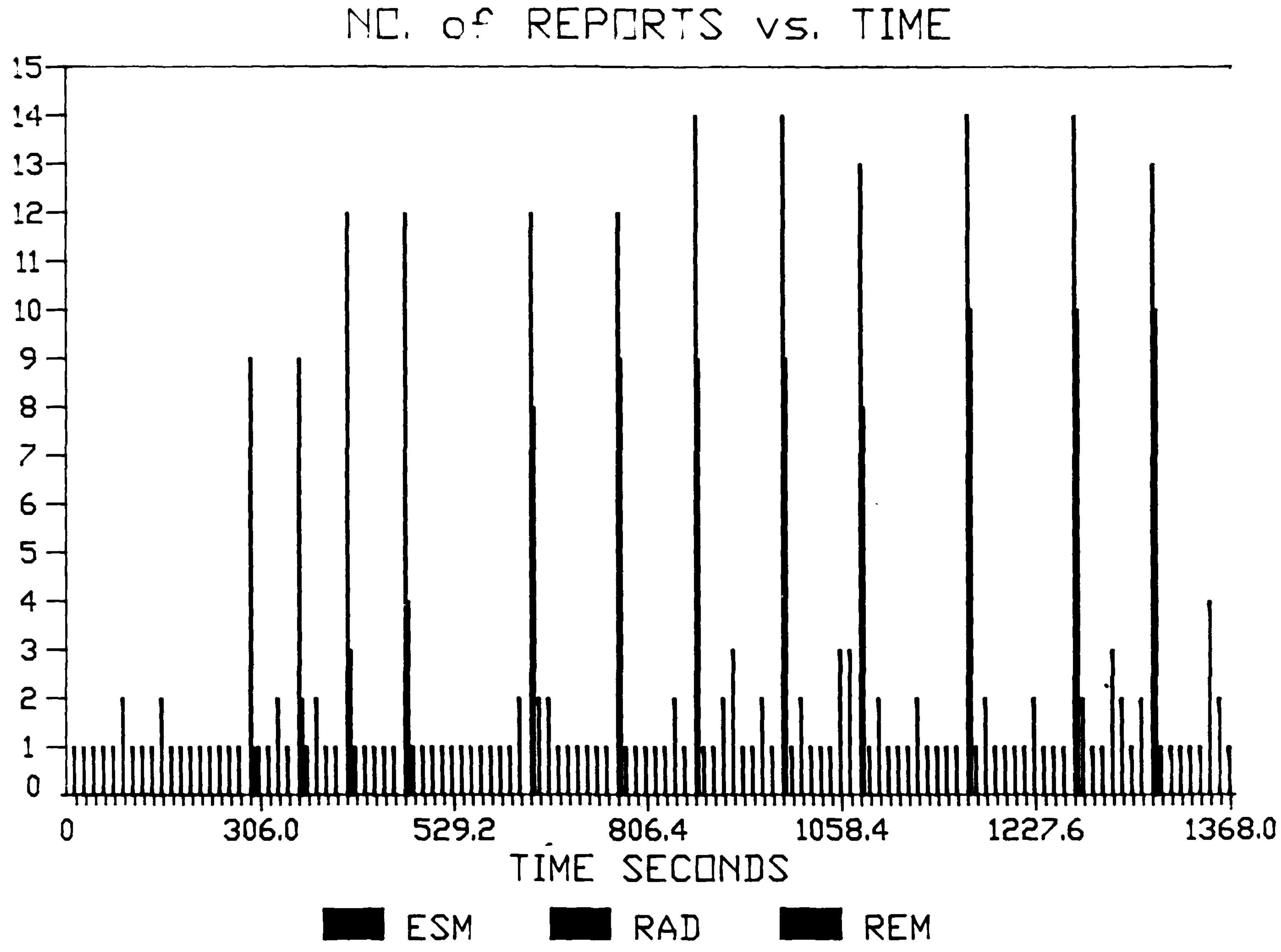


Figure 4. Number of reports versus time.

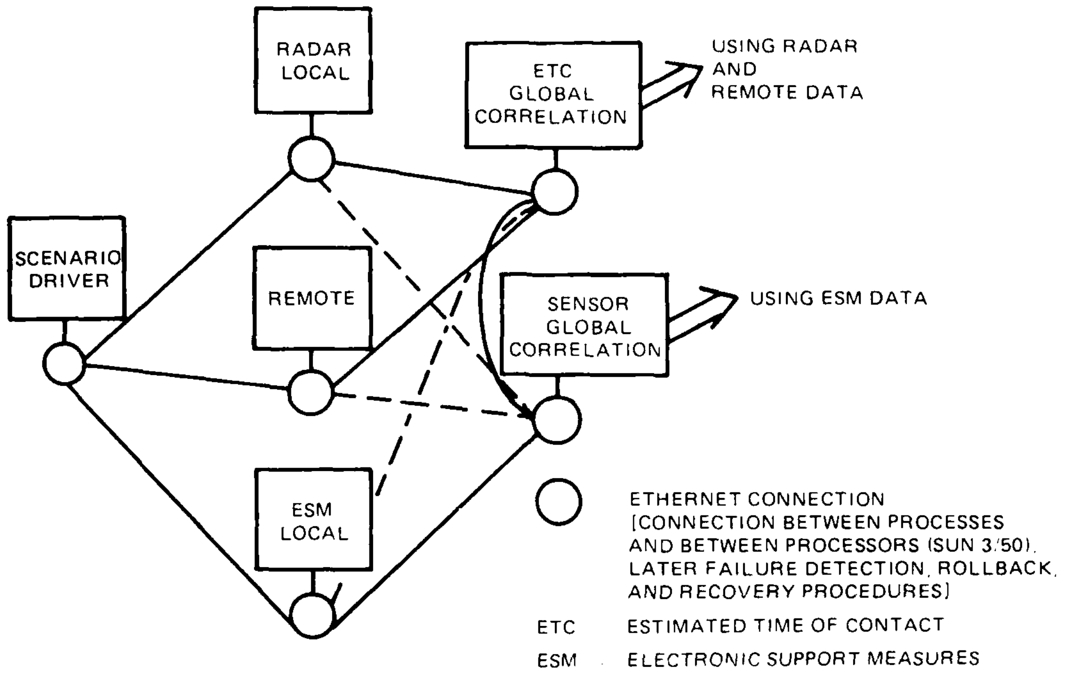

RADAR LOCAL
ETC GLOBAL CORRELATION
USING RADAR AND REMOTE DATA
SCENARIO DRIVER
REMOTE
SENSOR GLOBAL CORRELATION
USING ESM DATA
ESM LOCAL
ETHERNET CONNECTION [CONNECTION BETWEEN PROCESSES AND BETWEEN PROCESSORS (SUN 3/50). LATER FAILURE DETECTION, ROLLBACK, AND RECOVERY PROCEDURES]
ETC ESTIMATED TIME OF CONTACT
ESM ELECTRONIC SUPPORT MEASURES

Figure 5. Initial configuration for tracking model.

correlation process were used. Support software was provided by the Ethernet connection to smooth the interface to the UNIX pipes. The results of this experiment are shown in figure 6, and indicate that performance is improved statistically when more processing is performed concurrently at the local level. In this example, 18-percent reduction in processing at the global correlation level is shown when only hostile tracks, which are coming too close to own-ship coordinates, are correlated with a similar potential for improvement in threat assessment time.

$$1 - \frac{\text{hostile report time with too-close ETC}}{\text{all reports time with too-close ETC}} \times 100\% = \%\ \text{reduction}$$

Simulated report time = 648 seconds (relative)

	Radar	Remote	ETC Gobal Correlation
Processing Time ...	39%	35%	2%
Database Time ..	12%	Same	19%
Lock Time ...	44%	15%	—
Total Time ...	33%	12%	18%

Report % Correct	Quality % Incorrect	% Missed (No Correlation)
100%	0%	0%

Figure 6. Distributed data for initial configuration concurrency theory comparison.

What these results do not show is how well consistency of the assessment is maintained over a long period and how well performance is improved over that period. The preliminary results also need to be borne out by using more data as well as the extra processing necessary for electronic support measure local tracks and sensor global correlation. This will be carried out in 1988, along with a comparison with the centralized experiment (reference 7); see figure 7.

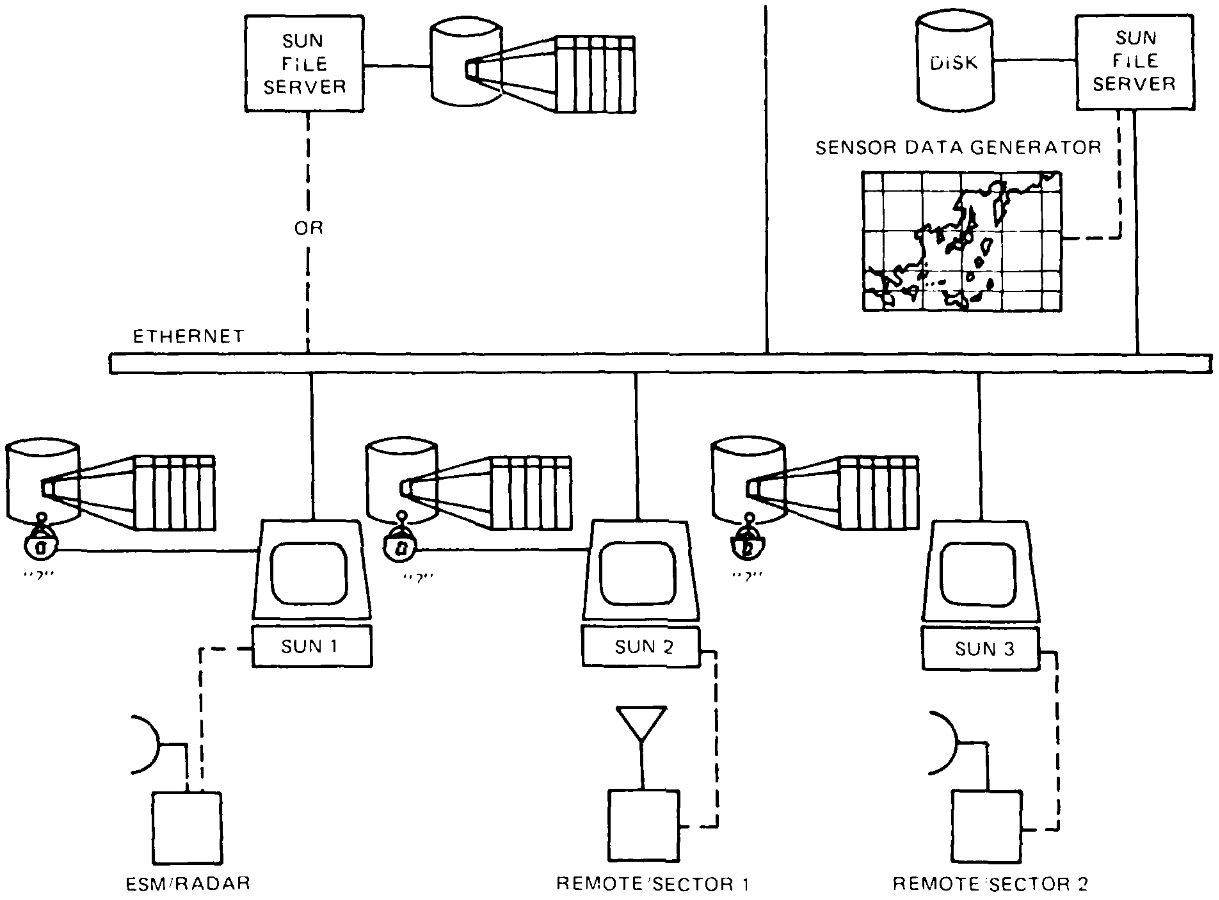


Figure 7. NOSC distributed tactical decision support tracking model testbed (using SUN processors).

DISTRIBUTED REAL-TIME DATABASE SYSTEM FOR TRACKING MODEL

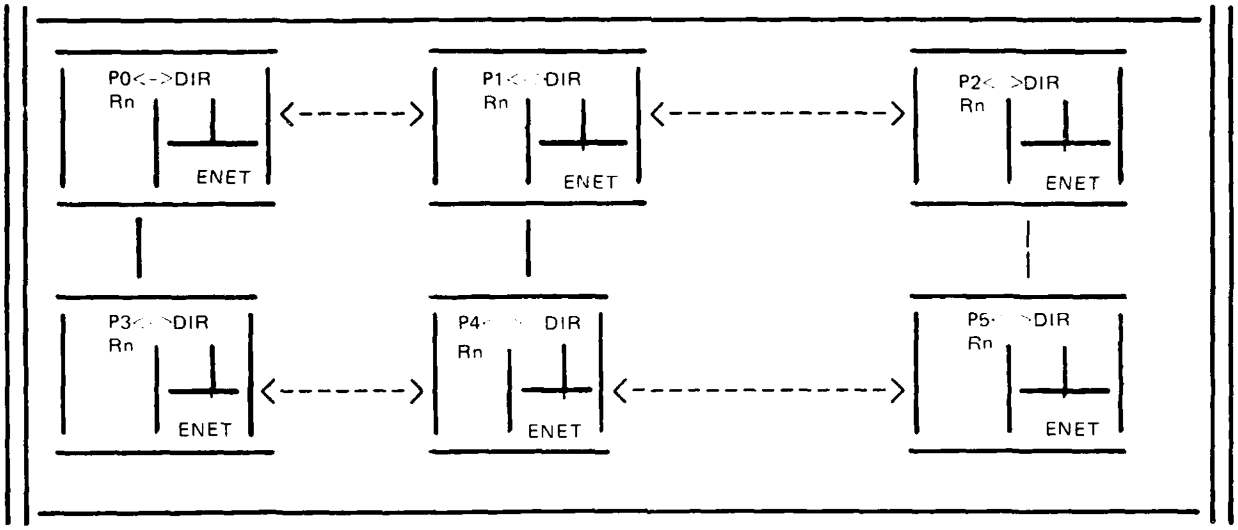
NOSC's research in improving real-time database management support for combat system and command control processing will first be involved in dynamically locating the data, rather than having their processor location fixed statically, as was the case in supporting our initial findings. This will include developing a module directory, as described in reference 8, and a corresponding rule set for management of the directory. This effort is described in more depth in reference 3. The directory (figure 8) will be maintained at each processing node, along with its best estimate of status information on each relation.

LOC	PROTECTION	USE DATA (IMPORTANCE)	LOCKED?	PROCESSING ACTION	TIME

Figure 8. Relation status directory.

All data required by the combat direction system analysis must be accessed through this module when invoking any relational command, such as join, restrict, project, list, update, delete, etc. Using the directory data residing in the module, relation data can be dynamically located to indicate which processor allocated the processing analysis, rather than have its processor location fixed statically. The directory will contain network node location of relations; protection key(s); use data according to importance (i.e., when should a current copy of a relation be maintained?); locking status; processing action occurring on the relation; and time of action. In the module, there will be a corresponding rule set of actions on the directory data that includes, for example, rules for when the relation can be accessed, updated and/or copied; analysis of a relation's locking status; rules to recover from the last consistent state of the relation; data projection rules; and management of currency as well as arrival of time-latent data.

Using this structure, the problems of handling data availability, node failure, lack of node availability because of overload, and timeliness (to be provided by the concurrency gained from FY87's research) will be smoothly handled. This methodology will be present in our tactical C^2 prototype testbed for use in experimentation with the theories developed in FY87 and FY88 research. In particular, such a real-time distributed database will facilitate the investigation of issues involved in maintaining consistent versions of distributed data and deciding which versions to use; i.e., more recent versions or the latest consistent version. This access is depicted in a generic sense in figure 9.



Pn = Process N-
each using rule structure

Rn = Relation N

DIR = Directory Module-
using rule structure

ENET = Ethernet Connection between
Processes and Processors

Figure 9. Generic model of real-time distributed database system.

OTHER POSSIBLE EXPERIMENTS

Preliminary findings suggest that correlation should be done only when a track is new or if it is behaving strangely at too close a range. The data were taken over a series of 50 reports for 648 simulated seconds, from a potential of 370 reports over a period of 1368 simulated seconds. In this first experiment, only the local radar process, the remote process, and the estimated-time-of-contact global kinematic correlation process for correlating radar and remote reports were used. Support software was provided by the Ethernet connection to smooth the interface to the UNIX pipes. These results indicate that performance is improved statistically when more processing occurs concurrently at the local level. In the example, 18-percent reduction in processing per track report at the global correlation level is shown when only hostile tracks that are coming too close to own-ship coordinates are correlated. We believe that a similar potential for improvement in threat assessment time would exist.

In the future, we plan to complete the implementation of new track report logic for electronic support measure (ESM) local track and sensor global correlation processing, using four Sun 3/50 processors with four Mbytes of main memory, supported by 337-Mbyte disk drives netted together via the Ethernet in our lab space. What these results will not show is how well the consistency of the assessment is maintained over a long period and how well performance is improved over that period. The preliminary results also need to be developed by using more data (with random noise inserted) and the extra processing required for ESM local tracks and sensor global correlation. This will be carried out in the future, as will the comparison with ground truth using the perfect data developed for the China Sea Scenario.

The next step will be to study methods for maintaining consistency of data versions and modes for processing failure, using our implementation of the real-time distributed database system. Throughout this phase of our research, we expect to work closely with CMU researchers on the proposed experiments. Our final report will show the research's relevance to systems such as those in ONT's distributed C^2 and multifunction C^2 workstation, and in a development program at NOSC entitled "Waterside Security."

REFERENCES

1. Sha, L., J.P. Lehoczky, and E.D. Jensen (all from Carnegie-Mellon University). 1985. *Modular Concurrency Control and Failure Recovery — Part I: Concurrency Control, and Part II, Failure Recovery.* Submitted for publication November 1985.

2. Small, D. 1984. *Machine-Based Information Systems for Navy C^2,* Proceedings of the 7th MIT/ONR Workshop on C^3 Systems, December 1984.

3. Small, D 1988. *Distributed Real-Time Database System for Command/Control and Combat Systems.* In preparation. Preliminary version available to qualified requesters.

4. Kauffold, R.P. 1986. *The Entity-Relationship Approach: A Good Tool for Tactical Data Systems?,* Thesis, Naval Postgraduate School, Monterey, CA, June 1986.

5. NAVSEA 0967-LP-8627D, *Advanced Combat Direction System (ACDS) Program System Specification for CV/CG Block 1 Combat Direction System (CDS),* 1984. October 1983, Change 2, June 1984.

6. Small, D. 1987. *Distributed Tactical Decision Support,* Proceedings of Joint Director of Laboratories, 1987, Command and Control Research Symposium, 16-18 June 1987. (Distributed September 1987.)

7. Small, D., L. Wong, and G. Sullivan. 1986. *Combat Systems Experiments (CSE) Centralized Modularity Experiments.* Naval Ocean Systems Center (NOSC) Technical Document 1029, September 1986.

8. Small, D., and W.W. Chu. 1979. *A Distributed Database Architecture for Data Processing in a Dynamic Environment,* IEEE Compcon 79, February 1979.

APPENDIX A:

COMBAT DIRECTION SYSTEM BUILDING-BLOCK DESCRIPTION SYNOPSIS

BY D. SMALL AND M. POHOSKI

A.1 SIMILAR SOURCE INTEGRATION (SSI) FUNCTIONS

Similar source integration (SSI) functions receive sensor data and develop intermediate track files. The combat direction system (CDS) contains four SSIs: radar/IFF, ESM, acoustic, and navigation. Each SSI receives sensor data from a similar type. The SSI combines the data to provide composite data for that sensor type. The radar/IFF SSI combines integrated automatic detection and tracking (IADT), target acquisition system (TAS), and identification friend/foe (IFF) data to maintain an intermediate track file. IFF ıransmissions are controlled manually or through surveillance control doctrine. The electronic support measures (ESM) and SSI process local and remote ESM data. The acoustic SSI function on aircraft carriers interfaces with the antisubmarine warfare (ASW) module function, which maintains its own intermediate track file. The navigation SSI function receives own-ship navigation data from the carrier navigation system (CVNS) and navigation data relative to own ship from the Joint Tactical Information Distribution System (JTIDS) terminal. The navigation (NAV) SSI compares the data and selects the best source for distribution within CDS. For purposes of this model, navigation will be assumed to be correct.

A.2 REMOTE DATA PROCESS (RDP) FUNCTION

The remote data process (RDP) function coordinates the receipt of remote track data and transmission of own-ship track data via the command and control processor (C^2P) to ensure a common CDS track file for all platforms.

A.3 DISSIMILAR SOURCE INTEGRATION (DSI) FUNCTION

The dissimilar source integration (DSI) function receives intermediate track files from the SSIs, RDP, surveillance control, and afloat correlation system (ACS), to be correlated into the CDS track file. ACS provides correlation recommendations between real-time and nonreal-time tracks. Each of the intermediate track files are updated as determined by sensor processing. DSI correlates and combines the dissimilar track files into a composite CDS track file. The CDS track file (TF) is extrapolated periodically to provide a current track database for use by all CDS functions. In addition, the CDS TF is duplicated and passed to interconnected systems for operator display and correlation recommendation processing.

A.4 DATABASE FUNCTION OF BATTLE FORCE TRACK FILE (BFTF)

The battle force track file (BFTF) will contain, as minimum, correlated track data (including identification wherever possible). A track in this context would have at least the attributes of location, course, speed, type, name (if available), identification, contact emissions, sensor type, track quality, time of arrival, current time, and end time. The database join key in the track file is track number and time of arrival. Supplemental to the track contents are included data supportive of the identification process, such as weapons on board various platforms, as well as the sensors available, the console operator's profile, and the data supportive of the mission planning process (e.g., information on threat profiles and own-ship weapon status).

A.5 MULTISOURCE IDENTIFICATION (MSID) FUNCTION

The multisource identification (MSID) function evaluates own-ship sensor and remote data to develop composite identification information.

A.6 SURVEILLANCE CONTROL FUNCTION

The surveillance control function provides for the control of own-ship sensor coverage. The operator is able to control the sensor operating parameters to match the own-ship task assignment such as inner air defense, ASW screen, etc. This control can be performed manually or accomplished automatically by CDS doctrine statements. If a sensor is not achieving the desired performance, an operator alert and recommended corrective action are provided. Manual and nonreal-time control are maintained through the surveillance control function. There is no attempt in this experimental model to consider manual adjustments for sensor performance.

A.7 MULTIWARFARE CONTROL (MWC) FUNCTION

The multiwarfare control (MWC) function will support the action of weapon assignment and subsequent scheduling and engagement. This action is based on threat evaluation data and mission planning data on threat profiles and weapon status. It will result in the control of weapon assets available on the carrier, including aircraft, and a variety of gun and missile systems on the various other ships.

A.8 MISSION CONTROL (MC) FUNCTION FROM WARFARE AREA

The mission control (MC) function supports integrated planning, resource allocation, and order execution. It includes effectiveness evaluation of operations in which the carrier is involved. It interfaces with, among other things, the flag data display system (FDDS), which supports the tactical flag command center aboard the carrier. It also provides a complete accurate assessment of the tactical situation.

A.9 COMBAT DIRECTION SYSTEM (CDS) OPERATOR CONSOLE BUILDING BLOCK

Each console will be provided automated capability for database management update, search, and analysis by using the relational join-and-restrict functionality. Scheduling of those functions will be provided, as will the scheduling of requests of other consoles or of the main processor for information such as track within radius of "x," hostiles in that radius, or changes in ID. Data used by the operator will be resident in the console processor's main memory, or can be sent on a predetermined basis or by operator request. Different profiles or contexts of expertise to be provided for the operator will be preselected for the console processor's memory, probably from battle force track file data.

APPENDIX B:

CHINA SEA SCENARIO SCRIPT

BY M. MIDDLETON
SYSTEM DEVELOPMENT CORPORATION
(NOSC CONTRACT N66001-83-D-0094)

During an emergency Central Intelligence Agency briefing to the President and Joint Chiefs of Staff, it was revealed that a significant seismic disturbance was detected at or near the Soviet Naval Base at Vladivostok. It is known that this base serves as a nuclear missile supply center for the Soviet Pacific Fleet. The CIA suspects that the detected disturbance might have been caused by an explosion at the base, and solicits the assistance of the military to substantiate their suspicions. All members of the Joint Chiefs of Staff unanimously agreed that verification of this information was of vital importance, since its validity would require a significant change in their immediate and near-future strategic plans. At the conclusion of the briefing, the President directed the Joint Chiefs of Staff to have their respective military organizations gather any and all intelligence they can relating to this hypothesis. In particular, the Chief of Naval Operations was ordered to have all naval platforms in the Western Pacific seek out any Soviet Navy vessels and monitor their performance for any erratic or unusual action.

Prior to receiving the CNO directive, two P3C aircraft were being dispatched on ASW barrier missions from the U.S. Navy base at Subic Bay. The USS CONSTELLATION (CV64) Battle Group was steaming south in the China Sea, where it was conducting training exercises. On receipt of the CNO message, the battle group commander ordered the launching of four CAPs to fan out to the north of his present position. In addition, one E2C was ordered on a reconnaissance mission 200 miles from the battle group on a bearing of 295 degrees, and another E2C on a similar mission 200 miles from the CONSTELLATION on a bearing of 25 degrees. Two surveillance missions were ordered to be conducted by S3As at distances of 150 and 200 miles from the CONSTELLATION, on bearings of 5 and 245 degrees, respectively. Two SURCAP missions were ordered to be located at a distance of 125 miles from the battle group on a bearing of 65 and 75 degrees, respectively. The two P3Cs departing from SUBIC were ordered to fly their assigned missions but to be prepared to receive redirection instructions.

The battle group commander ordered all surface platforms to immediately activate their surface search radars. All air combat patrols, reconnaissance, and surveillance aircraft were instructed to activate their respective radars after launch.

The battle group participating platforms were further ordered to report all radar or ESM sensor contacts immediately. On all NTDS-equipped platforms, the target-detection sensor information was to be forwarded to the CONSTELLATION via Link 11 channels.

In accordance with his newly assigned objectives, the battle group commander elected to order members of his group to keep their detected targets under surveillance even if it meant diverting from their present courses and speeds.

All tracking information obtained on hostile naval platforms and merchant ships was ordered to be forwarded to the CONSTELLATION until countermanding instructions were received. Abrupt changes in target course or speed were to be assigned top priority.

The scenario used to generate the CSE target track reports is referred to as the China Sea scenario. The composition of the China Sea scenario is given in the following table.

Quantity	Description
74	Friendly participating aircraft
11	Friendly participating surface platforms
1	Friendly land base
43	Hostile participating aircraft
23	Hostile participating surface platforms
7	Hostile land bases
5	Merchantman.

At the time the China Sea scenario was created, a set of initial conditions in the form of prestored orders was formulated. For this scenario, these orders pertained exclusively to the launching of aircraft from the participating aircraft carriers and the land base. Each order included the quantity of aircraft to be launched, the type of aircraft involved, the type of mission to be flown, and the ultimate on-station location from the launching platform. The launch time for each aircraft was automatically determined by the simulated flight operations function embedded within the event generator.

China Sea scenario commands are stored with the event generator. The file orders the friendly and hostile forces to activate their respective radar sets, and orders the friendly forces to activate their ESM devices. The contents of this file are given in appendix E.

Every tenth simulated engagement minute, the contents of the command file are read by the event generator program. The file contains command orders to launch aircraft from one of the participating aircraft carriers. The first commands are read at time 10 and the last at time 60. The information in the file is given in appendix E.

The China Sea scenario data were used by the event generator and its associated programs to obtain the target track information required to evaluate the correlator tracker algorithm. The target track data-gathering procedure was terminated after the 81st engagement minute. A total of 4483 new and updated track reports were generated. The distribution of these reports, according to their new or updated classification and detection sensor type, is as follows:

TYPE	NEW	UPDATE	TOTALS
ESM	17 (0.38%)	967 (21.57%)	984 (21.95%)
RADAR	84 (1.88%)	2213 (49.36%)	2297 (51.24%)
REMOTE	50 (1.11%)	1152 (25.70%)	1202 (26.61%)
TOTAL	151 (3.37%)	4332 (96.63%)	4483 (100.00%)

Each of the above data reports incorporates single-source integration. That is, all reports from a single source, such as radar, have been integrated so that each report is identified with a given track. A track (from a single source) is initiated by a new report and is subsequently modified by an update report with the assigned track number. Remote reports originate from platforms other than own platform. Remote reports may be generated by either radar or ESM sensors.

APPENDIX C:

ASYNCHRONOUS DATA GENERATION

Radar sensors serve as data input devices to NTDS. In general, they generate target data synchronously. Radars usually have a constant scan rate; i.e., same number of revolutions per minute. The time that a target blip is recorded on the radar screen is referred to as the time of incidence (TOI). When the input radar data are transmitted to a remote NTDS site via Link-11, they are subjected to certain delays. These delays are caused by the net control station polling function, which is heavily influenced by the number of NTDS ships connected to the Link-11 net and the amount of net traffic. There is a minimum delay (time$minimum) and a maximum delay (time$maximum) usually associated with each data transmission. Since the actual time that data are received at the remote NTDS site falls randomly between time$minimum and time$maximum, they are said to be received asynchronously. The time that the data are actually received at the remote NTDS site is referred to as the time of receipt (TOR).

To obtain the TOR time delays during the generation of tracking data from the IBGTT event generator, the following equation is used.

```
TOR=time$increment+[time$minimum+(time$maximum  time$minimum)*random]
```

where

time$increment	=	event time increment.
time$minimum	=	minimum simulated Link-11 transmission delays.
time$maximum	=	maximum simulated Link-11 transmission delays.
random	=	random number generator.

The TOR equation is used only when data are being transmitted to the designated own-ship platform from other platforms. Since radar and ESM data are immediately available on board the own-ship platform, no transmission delays are realized. These data are assumed to take place at the end of each game minute, whereas remote data occur randomly within each minute. All such data are interleaved with the command orders, as described in appendix E.

APPENDIX D:

CSE EVENT GENERATOR TARGET DETECTION DATA

The following data represent target detections produced by the radar and ESM models embedded in the event generator during execution of the China Sea Scenario for an 81-minute period. Data shown include local radar and ESM generated by own-ship and remote platforms.

MINUTE	LOCAL RADAR	LOCAL ESM	REMOTE RADAR	REMOTE ESM	TOTAL DETECTIONS
1	—	18	2	—	20
2	—	18	2		20
3	—	18	2	—	20
4	—	18	2	—	20
5	1	18	2		21
6	2	18	4	—	24
7	3	18	5	—	26
8	4	18	6	—	28
9	5	18	23	—	46
10	5	18	24	—	47
11	5	18	26	—	49
12	5	18	27	—	50
13	5	18	26	—	49
14	5	18	26		49
15	6	18	28	—	52
16	6	18	29		53
17	6	18	28		52
18	6	18	30	—	54
19	6	18	31		55
20	6	18	29	—	53
21	6	18	30		54
22	6	19	30		55
23	6	20	29		55
24	6	21	29	—	56
25	9	21	32	—	62
26	18	21	27	—	66
27	20	21	27	—	68
28	23	21	29		73
29	24	21	30		75
30	24	21	28	—	73
31	24	21	28	—	73
32	24	21	27	—	71
33	24	21	27	—	72
34	25	21	28	—	74
35	25	21	26	—	72
36	27	21	24	—	72

MINUTE	LOCAL RADAR	LOCAL ESM	REMOTE RADAR	REMOTE ESM	TOTAL DETECTIONS
37	26	21	24		71
38	27	21	25		73
39	29	20	24		70
40	29	21	22		72
41	30	21	20		71
42	31	21	18		70
43	29	21	18		68
44	32	22	19		73
45	35	22	22		79
46	35	22	23		80
47	36	22	26	1	85
48	35	22	27	1	85
49	37	22	29	1	89
50	39	22	32	1	94
51	41	22	34	1	98
52	43	22	36	1	102
53	45	29	23	1	98
54	46	29	26	1	102
55	47	28	25	1	101
56	48	29	27	1	105
57	50	29	27	3	109
58	49	29	29	3	110
59	47	29	30	3	109
60	46	31	17	4	98
61	46	31	16	3	96
62	46	31	16	3	96
63	46	31	16	3	96
64	48	31	18	2	99
65	50	31	19	2	102
66	52	31	21	1	105
67	54	31	21	1	107
68	56	31	21	1	109
69	57	31	22	1	111
70	58	31	22	1	112
71	60	31	23	1	115
72	59	31	24	1	115
73	58	31	23	1	113
74	54	31	23	1	109
75	54	31	23	1	109
76	54	31	21	1	107
77	53	31	20	1	105
78	52	31	19	1	103
79	49	31	18	1	99
80	48	31	16	1	96
81	46	31	16	1	94

APPENDIX E:

CSE EVENT GENERATOR STORED COMMANDS

The event generator command orders required to direct the activities of the various China Sea scenario participating units are described in the following paragraphs. The orders start the scenario and direct the various forces to activate their respective radar sets and ESM equipment. The command orders are listed below.

```
GO
FOR 1.1 ACTIVATE RADAR
FOR 1.1 ACTIVATE ESM
FOR 5.1 ACTIVATE RADAR
FOR 9.1 ACTIVATE RADAR
```

At simulated engagement minute 10, the orders below are submitted to the event generator program as input. These orders direct the aircraft carriers CONSTELLATION and KIEV to launch aircraft. The time of actual launch is determined by the simulated flight operations and depends on the type of aircraft and the flight-deck loading. The orders entered are shown below.

```
FOR KIEV LAUNCH 1 FORGR KAPA 0 400 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPE 210 350 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPB 45 400 10000
STOP
FOR DANANG LAUNCH 1 BEARD KAPC 60 450 25000
STOP
FOR KIEV LAUNCH 1 FORGR KAPD 30 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPE 60 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPH 90 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPG 105 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPF 120 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPJ 300 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPI 150 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPK 315 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPL 330 400 10000
STOP
```

The next set of command orders are submitted to the event generator program during simulated engagement minute 20.

```
FOR KIEV LAUNCH 1 FORGR KAPM 285 400 10000
STOP
FOR KIEV LAUNCH 1 FORGR KAPN 270 400 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPE 210 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPF 170 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPG 160 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPH 140 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPI 120 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPJ 82 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPK 200 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPL 180 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 F14A CAPM 330 350 10000
STOP
```

At simulated engagement minute 30, the following orders are next submitted to the event generator program as input.

```
FOR KIEV LAUNCH 1 HORMA KAPO 5 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPP 50 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPQ 35 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPR 65 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPS 95 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPT 110 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPU 125 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPV 305 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPW 155 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPX 320 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAPY 335 100 10000
STOP
```

```
FOR KIEV LAUNCH 1 HORMA KAPZ 290 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAP1 275 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAP2 355 100 10000
STOP
FOR KIEV LAUNCH 1 HORMA KAP3 40 100 10000
STOP
```

The following command orders are next submitted to the event generator program during simulated engagement minute 40.

```
FOR CONSTELLATION LAUNCH 1 F14A CAPN 0 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPO 355 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPP 350 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPQ 345 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPR 340 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPS 335 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPT 325 350 10000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPU 320 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPV 310 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPW 305 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPX 300 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPY 290 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAPZ 285 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP1 280 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP2 275 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP3 270 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP4 10 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP5 90 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP6 100 450 30000
STOP
```

```
FOR CONSTELLATION LAUNCH 1 FA18 CAP7 105 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP8 110 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP9 115 450 30000
STOP
```

At simulated engagement minute 50, the following orders are submitted to the event generator program as input.

```
FOR CONSTELLATION LAUNCH 1 FA18 CAP8 110 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CAP9 115 450 30000
STOP
FOR KIEV LAUNCH 1 HORMB KA19 345 100 4500
STOP
FOR KIEV LAUNCH 1 HORMB KA20 340 100 4500
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA10 175 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA11 165 300 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA12 155 300 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA13 150 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA14 145 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA15 135 450 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA16 130 400 30000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA17 125 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA18 15 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA19 20 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA20 75 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 FA18 CA21 85 400 20000
STOP
```

The following command orders stored are last submitted to the event generator program during simulated engagement minute 60.

```
FOR CONSTELLATION LAUNCH 1 SH3H CA22 225 80 3000
STOP
FOR CONSTELLATION LAUNCH 1 SH3H CA23 270 80 3000
STOP
```

```
FOR CONSTELLATION LAUNCH 1 SH3H CA24 315 80 3000
STOP
FOR CONSTELLATION LAUNCH 1 SH3H CA25 0 80 3000
STOP
FOR CONSTELLATION LAUNCH 1 SH3H CA26 45 80 3000
STOP
FOR CONSTELLATION LAUNCH 1 EA6B CA27 90 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 EA6B CA28 95 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 S3A CA29 295 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 S3A CA30 5 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 S3A CA31 170 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 S3A CA32 120 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 S3A CA33 25 400 20000
STOP
FOR CONSTELLATION LAUNCH 1 S3A CA34 95 400 28000
STOP
FOR CONSTELLATION LAUNCH 1 S3A CA35 315 400 27500
STOP
FOR CONSTELLATION LAUNCH 1 KA6D CA36 260 400 30000
STOP
FOR CONSTELLATION LAUNCH 1 KA6D CA37 30 400 26000
STOP
FOR CONSTELLATION LAUNCH 1 KA6D CA38 190 400 25000
STOP
```

APPENDIX F:

NEW TRACK REPORT LOGIC

Developed Jointly by D. Small, M. Trumpold, D. Brouhard, D. Schmidt, M. Pohoski, all of NOSC, and L. Sha and H. Tokuda of Carnegie Mellon University.

(As envisioned from an operator's console.)

All new reports entering the CV's combat direction system tracking model will be reported in the format of either an ESM local report, a radar local report, or a remote report, each with its unique local track number (a local report assumes the signal processing at the sensor level already has been accomplished). Each of these reports will be merged into the global track relation data structure as appropriate (i.e., the track is of high quality and meets integrity constraints). ESM reports will be added to the SENSOR global level of the track relation, and radar and remote reports will be added to the ETC global level of the track relation and, in each case, assigned a global track number. This new global track number will then be sent back to the local level as an indication of correlation with other tracks.

The following describes the logic that processes an incoming report at the local level and a subsequent global track level, as appropriate. A transaction is considered to be a correlation or association. Each track at each level of processing is considered to be an atomic data set in the Carnegie Mellon University sense of the phrase (reference 1). A transaction at a specific level is considered to be an elementary transaction.

I. RADAR LOCAL TRACK TRANSACTION FOR NEW TRACK REPORT FOR RADAR CONSOLE

1A. CONSTRAINTS SATISFACTION

Determine that the new report's quality (Q) is greater than 30%. If not, discontinue processing of the track report's logic and do not update local track history.

Calculate the new report's RANGE and SPEED and, based on that, calculate estimated time of contact (ETC). There is the assumption in the ETC calculation that COURSE will change such that the new report would be on a dead-reckoning worst-case track, coming directly at the own-ship current position. AZDIFF will indicate whether there has been any course variation since the last update on this track.

Determine that the new report meets ID (NTDS-CLASS) speed boundaries based on RANGE and ETC calculations derived according to the NTDS-MAX-CONSTRAINT relation (i.e., maximum speed constraints).

If the constraints are not satisfied, change NTDS class to unknown.

1B. RADAR ASSOCIATION TRANSACTION LOGIC

If this local RADAR track is "associated" with (i.e., has a global track number), update the local track history if there are no significant changes and do not send track up to the global level.

If the track is a new report, attempt the correlation of this report. If the track is not new, and if it is hostile and too close or has significant changes, attempt the correlation of this report. If the track is not new, and if it is friendly and has significant changes, attempt the correlation of this report. (Note: A friendly is not continually correlated if it is too close.) Correlation is attempted by using the higher level ETC GLOBAL TRACK TRANSACTION logic.

****

THERE IS LOCK POTENTIAL BETWEEN LOCAL TRACKS AND GLOBAL TRACKS, when using local versions at the local level, while the global track correlation is occurring. After correlation is completed, the new global track number must be communicated back to the local level, possibly including field updates (e.g., quality and ID changes) when local history is updated.

****

2. REMOTE ASSOCIATION TRANSACTION LOGIC FOR NEW TRACK REPORT FOR REMOTE CONSOLE

Calculate the new report's RANGE and SPEED and, based on that, calculate estimated time of contact (ETC). There is the assumption in the ETC calculation that COURSE will change such that the new report would be on a dead-reckoning worst-case track coming directly at the own-ship current position. AZDIFF will indicate whether there has been any course variation since the last update on this track.

If this local REMOTE track is "associated" with (i.e., has a global track number from OWN SHIP), update the local track history if there are no significant changes and do not send track up to next level.

If the track is a new report, attempt the correlation of this report. If the track is not new, and if it is hostile and too close or has significant changes, attempt the correlation of this report. If the track is not new, and if it is friendly and has significant changes, attempt the correlation of this report. (Note: A friendly is not continually correlated if it is too close.) Correlation is attempted by using the higher level ETC GLOBAL TRACK TRANSACTION logic.

****

LOCK POTENTIAL BETWEEN REMOTE TRACKS AND GLOBAL TRACKS, when using local versions at the local level, while the global track correlation is occurring. After correlation is completed, the new global track number must be communicated back to the local level, possibly including field updates (e.g., quality and ID changes) when local history is updated.

****

3. ESM LOCAL TRACK TRANSACTION FOR NEW TRACK REPORT FOR ESM CONSOLE

3A. CONSTRAINTS SATISFACTION

Determine that the new report's quality (Q) is greater than 30%. If not, discontinue processing and don't update local track history.

Determine that the new report meets ESM constraint for the track's postulated PLATFORM-KIND, using the PKIND-SEN-CONSTRAINT relation (i.e., sensor constraints of likely platforms).

Determine that the new report meets ESM constraint for NTDS-CLASS (ID) according to the NTDS-SEN-CONSTRAINT relation (i.e., sensor constraints of likely platforms).

If new report does not meet the constraints, put it in the do-not-meet-ESM-constraints file on disk.

3B. ESM ASSOCIATION TRANSACTION LOGIC

If this local ESM track is "associated" with (i.e., has a global track number), update the local track history if there are no significant changes and do not send it up to the global level.

If the track is a new report, attempt the correlation of this report. If the track is not new and if it is hostile and too close or has significant changes, attempt the correlation of this report. If the track is not new, and if it is friendly and has significant changes, attempt the correlation of this report. (Note: A friendly is not continually correlated if it is too close.) Correlation is attempted by using the higher-level SENSOR GLOBAL TRACK TRANSACTION logic.

THERE IS LOCK POTENTIAL BETWEEN LOCAL TRACKS AND GLOBAL TRACKS, using local versions at the local level, while the global track correlation is occurring. After correlation is completed, the new global track number must be communicated back to the local level, possibly including field updates (e.g., quality and ID changes) when local history is updated.

4. ETC (ESTIMATED TIME OF CONTACT) GLOBAL TRACK CORRELATION TRANSACTION FOR NEW TRACK REPORT

If the new report is outside of sector, communicate the report information to the console operator handling that sector.

If the track has significant changes and has a global track number, go directly to 4B.

4A. If the new report track quality (Q) > = 90% represents a confirmed friendly track, assign a new global track number and send back to the local track level releasing lock potential.

If the new report represents a confirmed hostile track with new report quality (Q) > = 90%, and if the RANGE is too close or ETC is too soon,

request verification from other operators, potentially locking the track information until action is agreed on.

If they agree, notify the tactical action officer (TAO) and weapon control, and send all known track information to weapon control for engagement,

release lock on the track information and send it back to the local track level as well.

If the TAO and/or weapon control decide not to engage, all known track information is returned with changes, if any, noted.

4B1. If this track has a global track number, check track history. If the ID disagrees and track quality is good, delete track history for this track because it is now different and go through the remaining logic to resolve differences. If ID disagrees and track quality < 90%, go through the remaining track logic to resolve differences. Otherwise, the track logic processing for this track is done.

Check NTDS-CLASS for a match between the tracks. If there is no match, investigate the next track.

If the two tracks being compared are RAD and ESM (or REM) tracks, check the PKIND-MAX-CONSTRAINT relation to determine whether SPEED is valid for PLATFORM-KIND. If not, discontinue processing of track.

If the two tracks being compared are RAD and ESM, check current tracks for those with the same approximate RANGE and ETC (plus or minus a differential with respect to time; i.e., the older the report the greater the differential).

Check matches for those with the same approximate COURSE (plus or minus a differential) to further eliminate possible tracks.

4B2. If more than one (1) track matches, check global track history on all matches for radical RANGE/ETC/COURSE changes, with the suggestion of possible ID changes. If no ID changes are suggested, continue to go back in track history.

If still no resolution as to the number of matches, back up one time point and repeat the above checks for all tracks, using an extrapolation forward, or dead-reckoning, in time to match with new track report's time.

If there is a match, update the old track that was matched. If track hostile and ETC (weapon of track's platform or track) are too close when the WEAPON SUBTRANSACTION is called, ask operators for help and inform the TAO.

Insert or merge [depending on whether time of arrival (TOA) is same or not] the new copy with the latest information, change the new report track number to the old one that was matched, push down the stack of global track history, and

****

return the result to the local track level, releasing the lock potential.

****

4B3. Else, if no tracks match, request help from other operators to identify (ID) the new report. If other operators can identify the report, add their results as a new track in the console's global ETC track relation. If the track is hostile and ETC for track or weapon of track's platform when the WEAPON SUBTRANSACTION called is too close, inform TAO, then

****

insert results as a new track in the console's global ETC track relation, informing TAO if the weapon's ETC of track's platform or track's ETC are too close. Return global track number and attribute changes to local track level and release lock potential.

****

If there is still more than one match, record each match as a copy of the track that was matched, push down the global track history on each old track, and

****

return the global track number and attribute changes down to the local track level, releasing lock potential.

****

If there are no matches,

****

the global operator can release the lock and send the new global track number down to the local level.

****

5. SENSOR GLOBAL TRACK CORRELATION TRANSACTION FOR NEW TRACK REPORT

Check current tracks in the SENSOR track file for those with the same SENSORS and same ID (NTDS-CLASS), COURSE, and PLATFORM-KIND.

If more than one (1) track matches, back up one time point, repeating the above checks for all tracks until there is only one track that matches or the last track data points are reached. If more than one matches, record each match as a copy of the track that was matched, push down the track on the global track history stack, and send the global track number and attribute changes down to local level.

****

LOCK POTENTIAL between ETC and SENSOR GLOBAL CORRELATION TRANSACTIONS if not on same node.

****

If there is no match, put the new track copy into the console's global track sensor database and send the new global track number down to local level.

6. WEAPON SUBTRANSACTION OF ETC GLOBAL CORRELATION TRANSACTION FOR NEW TRACK REPORT

Parameters passed to this subtransaction include track numbers, TOA, ID = hostile, new report quality (Q) (which will be greater than 30%), and weapon range and speed for the platform(s) of the track relation.

For resulting SENSOR(s), determine PLATFORM-NAME(s) on which SENSOR is located in SHIP-SENSOR or AIRCRAFT-SENSOR relation.

For each PLATFORM-NAME, determine PLATFORM-CLASS from which WEAPON-NAME(s) on the PLATFORM can be determined from SHIPWEPS relation.

Determine the WEAPON which has the greatest MAXRANGE and WEAPON SPEED.

For each track number sent to this transaction, calculate weapon ETC based on maximum weapon MAXRANGE and weapon SPEED for its platform and *range to* OWN SHIP.

Inform ETC GLOBAL TRACK TRANSACTION of results of track matching, including possible new threat potential.

7. INTERPROCESS COMMUNICATIONS (OR ETHERNET CONNECTION) FOR THE DISTRIBUTED TRACKING MODEL

The interprocess communications (COMMS) process for the distributed tracking model (DTM) is accomplished by using DoD's TCP/IP standard under Sun Microsystems UNIX 3.2, a reliable communications link between multiple processors.

The COMMS process created for DTM is meant to be able to handle the range of conditions that occur to cooperating processes without adversely affecting the throughput of data to healthy processes. Death, restart, I/O blocking, and buffering are all handled independently for each connection.

The data flow through the COMMS process follows a rather simple path. COMMS constantly loops, looking at all the connections, checking to determine whether any input is pending. If data are present, they are read and an attempt is made to pass the track up a pipe to the database process. If the pipe is blocked because of the buffered I/O already waiting in the pipe, the track is placed in a write queue. The track in the write queue is written to the pipe as soon as the pipe is no longer blocked. The flow of tracks out of COMMS takes a similar approach. Tracks are read in from the pipe as soon as they appear. If the destination of this track is up, an attempt is made to write the track to the remote process. If the connection is blocked with buffered I/O on the net, the track is placed in a write queue for the network. The same processing also occurs if the remote process is dead and the track is buffered in a write queue. When COMMS is doing nothing, and all its queues are empty, it sleeps until I O interrupts it with an asynchronous trap.

Presently, the COMMS process does not attempt to limit the growth of these queues, although this should be done to prevent a permanently dead process from choking all cooperating processes with dead tracks. In the future, an expert system could decide that this "dead" processor is not coming back, and reassign its duties to some other machine. In this case, the queued tracks could be passed over to the newly designated processor.